ADAPTIVE CONTROL OF THIRD HARMONIC GENERATION VIA GENETIC ALGORITHM

A Thesis

by

XIA HUA

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

August 2010

Major Subject: Physics

ADAPTIVE CONTROL OF THIRD HARMONIC GENERATION VIA GENETIC ALGORITHM

A Thesis

by

XIA HUA

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved by:

Chair of Committee,	Alexei V. Sokolov
Committee Members,	M. Suhail Zubairy
	George R. Welch
	Robert D. Nevels
Head of Department,	Edward Fry

August 2010

Major Subject: Physics

ABSTRACT

Adaptive Control of Third Harmonic Generation via Genetic Algorithm. (August 2010)

Xia Hua, B.S., Wuhan University; B.Eng., Huazhong University of Science & Technology

Chair of Advisory Committee: Dr. Alexei V. Sokolov

Genetic algorithm is often used to find the global optimum in a multi-dimensional search problem. Inspired by the natural evolution process, this algorithm employs three reproduction strategies -- cloning, crossover and mutation -- combined with selection, to improve the population as the evolution progresses from generation to generation.

Femtosecond laser pulse tailoring, with the use of a pulse shaper, has become an important technology which enables applications in femtochemistry, micromachining and surgery, nonlinear microscopy, and telecommunications. Since a particular pulse shape corresponds to a point in a highly-dimensional parameter space, genetic algorithm is a popular technique for optimal pulse shape control in femtosecond laser experiments.

We use genetic algorithm to optimize third harmonic generation (THG), and investigate various pulse shaper options. We test our setup by running the experiment with varied initial conditions and study factors that affect convergence of the algorithm to the optimal pulse shape. Our next step is to use the same setup to control coherent anti-Stocks Raman scattering.

The results show that the THG signal has been enhanced.

ACKNOWLEDGEMENTS

I would like to thank my committee chair, Dr. Sokolov, and my committee members, Dr. Zubairy, Dr. Welch, Dr. Nevels, for their guidance and support throughout the course of this research.

Thanks also go to my friends and colleagues, Xi Wang and Steve Scully. They gave me lots of help in completing the experiment and programming.

Finally, thanks to my mother, Huilan Xu, and father, Lianhong Hua, for their encouragement and to my wife, Yin Hong, for her patience and love and to my baby girl, Christina Tzu-Hsin Hua.

NOMENCLATURE

AO-PDF	Acousto-optic programmable dispersive filter
CARS	Coherent anti-stocks raman scattering
FROG	Frequency resolved optical gating
FWHM	Full width at half maximum
GA	Genetic algorithm
SHG	Second harmonic generation
THG	Third harmonic generation

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE Page

1. INTRODUCTION

1.1 Scheme of the genetic algorithm controlled experiment

Chemical or physical control using femtosecond pulse shaping has been recently studied by several groups [1-8]. The common experimental scheme includes a self learning loop composed of an ultrashort laser, a femtosecond pulse shaper, and a computer running an evolutionary algorithm.

The ultrashort laser pulse can be shaped by computer controlled femtosecond pulse shaper which adjusts spectral amplitudes and phases. The shaped laser pulse can then be used to excite a sample, and we can measure some response of the sample, such as second harmonic generation (SHG) [2], third harmonic generation (THG), coherent anti-Stocks raman scattering (CARS) signal [2, 7, 8]. The measured signal is then used as the feedback for the genetic algorithm, and is called the fitness value. The genetic algorithm can generate the next generation which is better than the previous one. This process is then repeated several times, until the fitness value converges to an optimum (Fig.1.).

This thesis follows the style of Physical Review Letters.

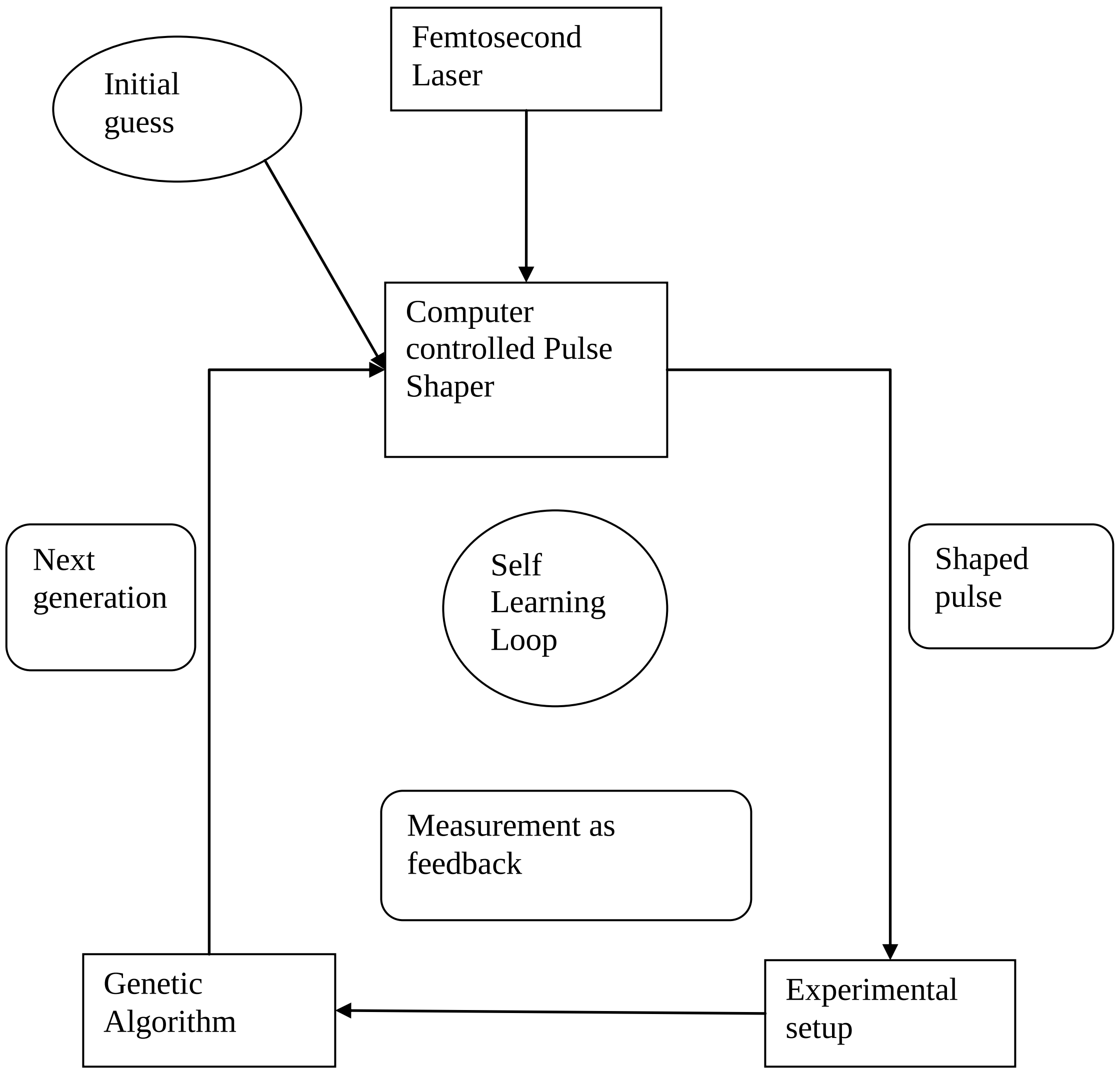

Fig. 1. The scheme of the self learning loop in a coherently controlled, genetic algorithm optimized, femtosecond laser experiment. The loop begins with an initial guess. Then the shaped pulse interacts with the sample. A suitable measurement is done. Then the measurement result is used as the feedback for the genetic algorithm to produce the next generation, which is better than the previous generation

1.2 Advantages of the genetic algorithm

Since no detailed knowledge of the experimental system is needed, the genetic algorithm can be applied to systems with a large number of control parameters, provided that the system can give a quantified feedback. In the meantime, because the molecular Hamiltonians are complicated and sometimes the Hamiltonians are unknown, theoretical calculations of quantum control experiments usually have large errors. Genetic algorithm is extremely useful for such quantum control experiments.

2. GENETIC ALGORITHM

2.1 What is genetic algorithm?

Genetic Algorithms are search algorithms that mimic natural selection and natural genetics. [16] They use a string structure called a vector $\vec{x}$ to represent each search point in the search space.

$$\vec{x} = (x_1, x_2, \cdots x_N) .$$

Then one population is a set of M such vectors

$$\vec{x}_j \in R^N , \; j = 1, \cdots, M .$$

We call this population M .

There is a fitness function f mapping each string structure to a fitness value.

$$\text{Fitness value} = f(\vec{x})$$

A genetic algorithm searches for a point with the best fitness value. The algorithm begins with an initial generation, which is generated either randomly or by taking a good guess. Then it generates the next generation by three methods. The string structures with best fitness values would be selected as the parents of the next generation. This process is called selection. These parents would create the next generation by cloning, recombination and mutation (Fig.2).The new generation created would have better fitness value than the previous generation. The selection and creation process continues for several generations. Finally, the fitness value would converge; the vector structure corresponding to the best fitness value is the optimized result we can get from the genetic algorithm.

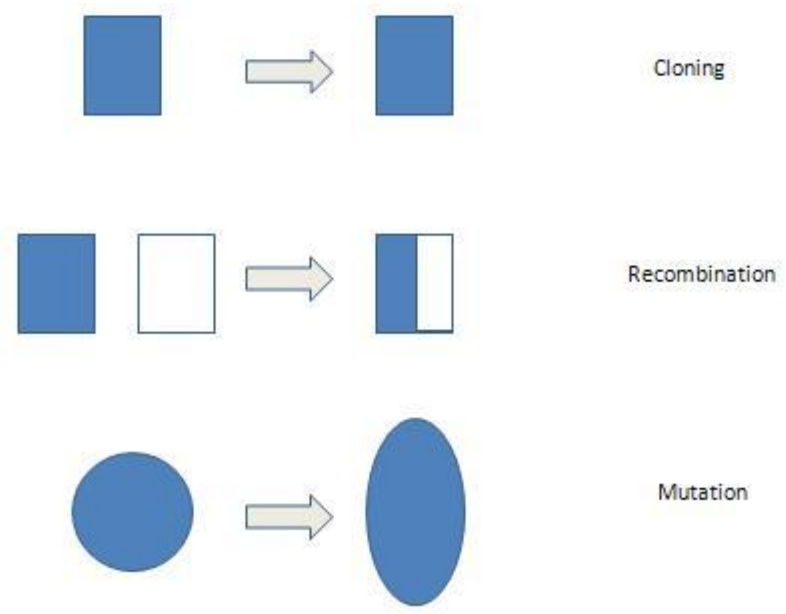

Fig. 2. An illustration of the cloning, recombination and mutation process

2.2 Selection/cloning

The process of the selection is to select the individuals with the best fitness value. These individuals can serve as the parents of the next generation. They can produce children with better fitness value. To ensure that, the next generation is not worse than their parents, the simplest way is to clone selected individual with the best fitness value to the next generation.

2.3 Recombination

Recombination is the process that uses two parents $\vec{x_i}, \vec{x_j}$ to generate one child. The two parents are chosen randomly from the selected parents with the best fitness value. There are several methods used to create a recombination child.

2.3.1 Single-point crossover

An index k' was chosen randomly with $1 \le k' \le N$. The child $\vec{y}$ then was created as

$$y_k = \begin{cases} (\vec{x_i})_k, & 1 \le k \le k' \\ (\vec{x_j})_k, & k' < k \le N \end{cases}$$

2.3.2 Two-point crossover

Two indexes k', k'' was chosen randomly with $1 \le k' \le k'' \le N$. The child $\vec{y}$ then was created as

$$y_k = \begin{cases} (\vec{x_i})_k, & 1 \le k \le k' \\ (\vec{x_j})_k, & k' < k < k'' \\ (\vec{x_i})_k, & k'' \le k \le N \end{cases}$$

2.3.3 Multiple crossover

A sequence $\vec{r}$ was selected randomly, where $r_k \in \{0,1\}, k = 1, \cdots, N$ with probability $P(r_k = 0) = P(r_k = 1) = \frac{1}{2}$. The child $\vec{y}$ then was created as

$$y_k = \begin{cases} (\vec{x_i})_k, & r_k = 0 \\ (\vec{x_j})_k, & r_k = 1 \end{cases}$$

2.3.4 Intermediate recombination

The child $\vec{y}$ then was created as

$$\vec{y} = \frac{\vec{x_i} + \vec{x_j}}{2}$$

2.4 Mutation

Mutation is the process where we add a small variation to the vector elements. We need to construct a sequence $\vec{r}$ randomly, where $r_k \in \{0,1\}, k = 1, \cdots, N$ with probability $P(r_k = 1) = P_{mut}, P(r_k = 0) = 1 - P_{mut}$. P_{mut} is called the mutation rate. The child $\vec{y}$ then was created as

$$y_k = \begin{cases} x_k + \sigma m_k, & r_k = 1 \\ x_k, & r_k = 0 \end{cases},$$

where σ is the step length, by which the amount of change caused by mutation is determined. m_k is a random number with Gaussian probability distribution around zero,

$$P(m_k) = \frac{1}{\sqrt{2\pi}} e^{(-m_k^2/2)} .$$

σ is vitally important to the convergence speed of the algorithm. In some situations, the mutation can produce many better children than the previous generation. It means that the genetic algorithm is searching in the parameter space where mutation can produce better fitness values. In this case, we can increase the step length to speed the convergence up. In other situations, the mutation fails to produce many better children compared to the previous generation. It means that the mutation is ineffective. This may indicate that an optimum has been already approached. In this situation, the step length must be decreased.

We introduce the ratio of successful mutation to the population:

$$\eta = \frac{n_{suc}}{n_{tot}} \leq 1 ,$$

where n_{suc} is the number of the successful individual created by mutation in the $t-1$ generation. n_{tot} is the population. The step length of generation t is given by:

$$\sigma_t = \begin{cases} \sigma_{t-1} q & \eta \leq \eta_c \\ \sigma_{t-1} / q & \eta > \eta_c \end{cases} ,$$

where $0 \leq \eta_c \leq 1$ and $0 < q < 1$. When $\eta \leq \eta_c$, that means the mutation is failing to produce better individuals. So we multiply a contraction factor q to decrease the step length. When $\eta > \eta_c$,that means the mutation successfully produces better individuals. So we divide by the contraction factor q to increase the step length.

3. EXPERIMENT

3.1 Experimental setup

The experimental setup is shown in Fig 3. We use Coherent ultra-fast laser system as the light source. The femtosecond laser has a repetition rate at 1 KHz, and power of 1 W. The center wavelength of the laser pulse is 800 nm. The laser beam goes through a neural density filter, so that the pulse shaper can be protected from high laser intensity. We use Fastlite Company's Dazzler ultrafast pulse shaper, which uses an acousto-optic programmable dispersive filter (AO-PDF) realized by TeO_2 crystal [17], to shape the femtosecond laser pulse. The shaped pulse is then focused on a thin glass to produce THG. The produced third harmonic signal at 266 nm can be observed by the Ocean Optics USB2000 spectrometer.

We can use frequency-resolved optical gating (FROG) to measure the duration of the ultra-short laser pulse. Four SF11 glass pieces are removable. They can be used to add higher order chirp to the laser pulse.

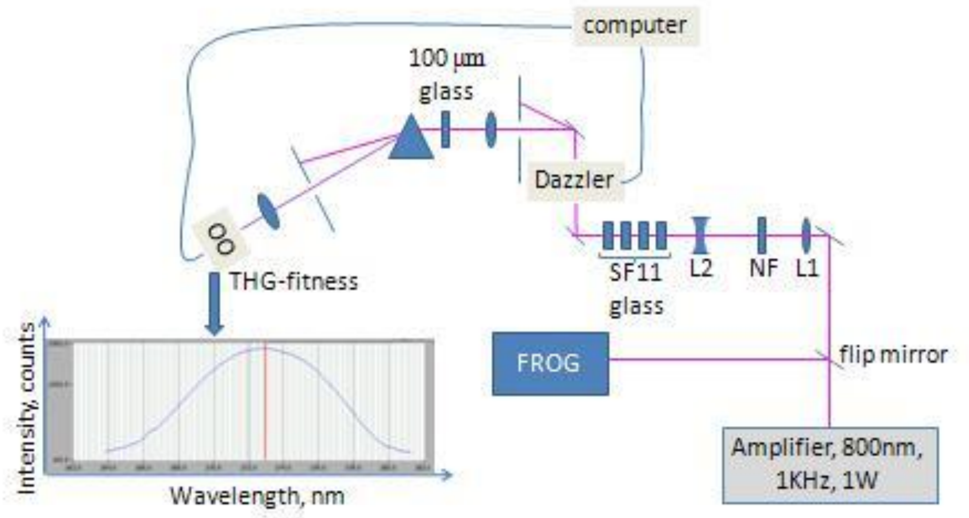

Fig. 3. The third harmonic generation experiment setup. The average intensity of THG in the wavelength range 264 -282 nm is the fitness value

The Dazzler is controlled by a computer running the genetic algorithm. The ultrafast pulse is controlled by adjusting different parameters using the Dazzler. These parameters can serve to define individuals for the genetic algorithm. The Ocean Optics USB2000 spectrometer can measure the average intensity of the THG at 266nm. The intensity serves as the fitness value for the genetic algorithm. Then the genetic algorithm runs in the search space produced by the parameters to find the optimized THG intensity.

3.2 Experimental procedure

We keep the amplitude of the input pulse constant. The amplitude is calculated by

$$A_{dial}(\omega) = f(\omega) \cdot g(\omega),$$

where

$$f(\omega) = \exp[-((\omega - \omega_0)/\delta\omega_0)^6],$$

with

$$\omega_0 = 2\pi c / \lambda_0,$$

$$\chi_0 = \delta\lambda_0 / (2\lambda_0),$$

$$\delta\omega_0 = \omega_0 \cdot (\chi_0 - \chi_0{}^3),$$

λ_0 = position,

$\delta\lambda_0$ = width, and

$$g(\omega) = 1 - k \cdot \exp[-((\omega - \omega_1)/\delta\omega_1)^2],$$

with

$$\omega_1 = 2\pi c / \lambda_1,$$

$$\chi_1 = \delta\lambda_1 / (2\lambda_1),$$

$$\delta\omega_1 = \omega_1 \cdot (\chi_1 - \chi_1{}^3)/2,$$

λ_1 = hole position,

$\delta\lambda_1$ = hole width,

k = hole depth .

The quantities $\lambda_0 = \text{position}$, which is the center wavelength of the pulse; $\delta\lambda_0 = \text{width}$, which is the full width at half maximum (FWHM) of the spectrum, can be set at the control panel of the Dazzler software. We also can make a hole in the input spectrum, which is described by the hole position, hole width and hole depth. If we don't need to make the hold, we set the hole position equal to the center wavelength of the pulse, and the hole width and hole depth both equal to zero.

The phase of the shaped pulse can be calculated by

$$\phi_{dial}(\omega) = -(a_1 \cdot (\omega - \omega_0) + \frac{a_2}{2} \cdot (\omega - \omega_0)^2 + \frac{a_3}{6} \cdot (\omega - \omega_0)^3 + \frac{a_4}{24} \cdot (\omega - \omega_0)^4),$$

where

a_1 = delay ,

a_2 = second order phase coefficient,which is chirp ,

a_3 = third order phase coefficient,which is quadratic chirp ,

a_4 = fourth order phase coefficient,which is cubic chirp .

There are two methods to control the phase of the shaped pulse. One is by setting phase corresponding to each wavelength. The other way is by setting the parameters a_1, a_2, a_3, a_4 and calculating the phase function by $\phi_{dial}(\omega)$. In other words, using the first method, genetic algorithm would control the phase directly. Using the second method, genetic algorithm would control the four parameters a_1, a_2, a_3, a_4. We would test these two control methods in the rest of the paper.

3.2.1 Manually find the optimized polynomial coefficients of phase function

Before we start using the genetic algorithm, we need to manually find the best polynomial coefficients of phase function, which controls the Dazzler to produce the strongest THG signal. We keep the amplitude of the laser pulse constant throughout the whole experiment. We set the center wavelength at 805 nm. The FWHM for the amplitude is 80 nm. We change the delay of the phase to 3500 fs and chirp to 15000 fs^2. The maximum intensity of THG we can get the manually is 1000 counts.

3.2.2 Genetic algorithm for the wavelength control

In this part of the experiment, we let the genetic algorithm deal with the phase directly. We set phases corresponding to each wavelength in the range from 765 nm to 845 nm, with the step of 3 nm. These phases form a string structure of 27 variables, which we use as the individual of the genetic algorithm. The genetic algorithm produces 30 such individuals per generation. We measure the THG signal corresponding to each phases in one generation using Ocean Optics USB2000 spectrometer. The integration time of the spectrometer is 400 ms and the measurement region is from 264 nm to 282 nm. The intensity of the THG is used as the feedback of the genetic algorithm (Fig.4). If not specified, the genetic algorithm is running with the variation of 0.6.

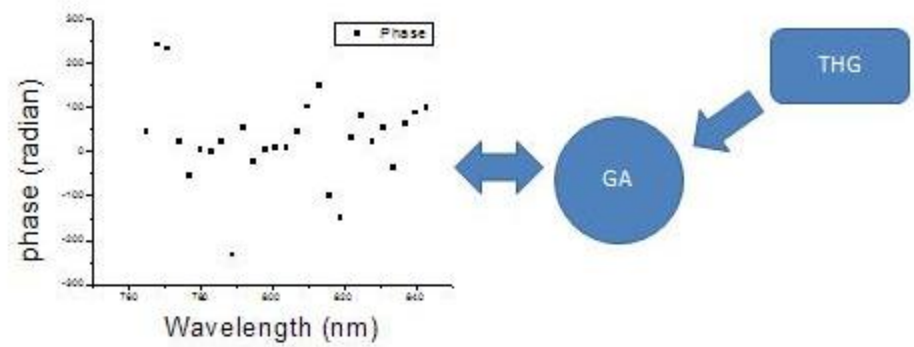

Fig. 4. A scheme for the genetic algorithm running in wavelength-control mode

3.2.2.1 Test of the genetic algorithm for different initial phases

We run the genetic algorithm several times. The initial generation is produced by a random function. But we insert some individuals which can produce good THG signals, so that the genetic algorithm can have some signal to optimize.

We call such inserted individuals are seeds. They can be calculated by the phase function we have given before.

$$\phi_{dial}(\omega) = -(a_1 \cdot (\omega-\omega_0) + \frac{a_2}{2} \cdot (\omega-\omega_0)^2 + \frac{a_3}{6} \cdot (\omega-\omega_0)^3 + \frac{a_4}{24} \cdot (\omega-\omega_0)^4) .$$

We keep the delay $a_1 = 3500 \text{ fs}$.

In the groups (a) and (b), we give one seed; we only set the chirp $a_2 = 15000 \text{ fs}^2$. The quadratic chirp and cubic chirp are equal to zero.

In the group (c), we give five seeds. Their parameters are (1) chirp $a_2 = 13500 \text{ fs}^2$, (2) chirp $a_2 = 16500 \text{ fs}^2$, (3) chirp $a_2 = 15000 \text{ fs}^2$, quadratic chirp

$a_3 = 5\times10^4$ fs^3 (4) chirp $a_2 = 15000$ fs^2 , quadratic chirp $a_3 = -5\times10^4$ fs^3 (5) chirp $a_2 = 18000$ fs^2 .

In the group (d), we give five seeds again. Their parameters are (1) chirp $a_2 = 10000$ fs^2 , (2) chirp $a_2 = 20000$ fs^2 , (3) chirp $a_2 = 15000$ fs^2 , quadratic chirp $a_3 = 5\times10^4$ fs^3 (4) chirp $a_2 = 15000$ fs^2 , quadratic chirp $a_3 = -5\times10^4$ fs^3 (5) chirp $a_2 = 15000$ fs^2 , cubic chirp $a_2 = 1\times10^7$ fs^4 .

Group (e) is the same as group (d), except the variation is 1.2.

Group (f) is the same as group (d), except the variation is 1.8.

In group (g), we only give one seed, with the chirp $a_2 = 16500$ fs^2 .

All these groups cannot give us satisfactory results (Fig.5). Groups (a) and (b) have very limited increase in the THG signal, because they begin from a point which already has a good THG signal. Groups (c) and (g) have some increase in the THG signal, but the improvement is not satisfactory. Groups (d), (e) and (f) don't have any improvement in the THG signal. The experiment is a failure in the wavelength control.

The possible reasons why the genetic algorithm fails in this part of the experiment is that: (1) There may be a large amount of noise here, so that genetic algorithm cannot compare the fitness values of any two measurements. (2) The THG signal responds best to a continuous phase function; what we give here is a descrete phase function.

We also learn from this part of the experiment, that the genetic algorithm needs some THG signal to start optimization. Otherwise, the genetic algorithm cannot give any improvement for the THG signal.

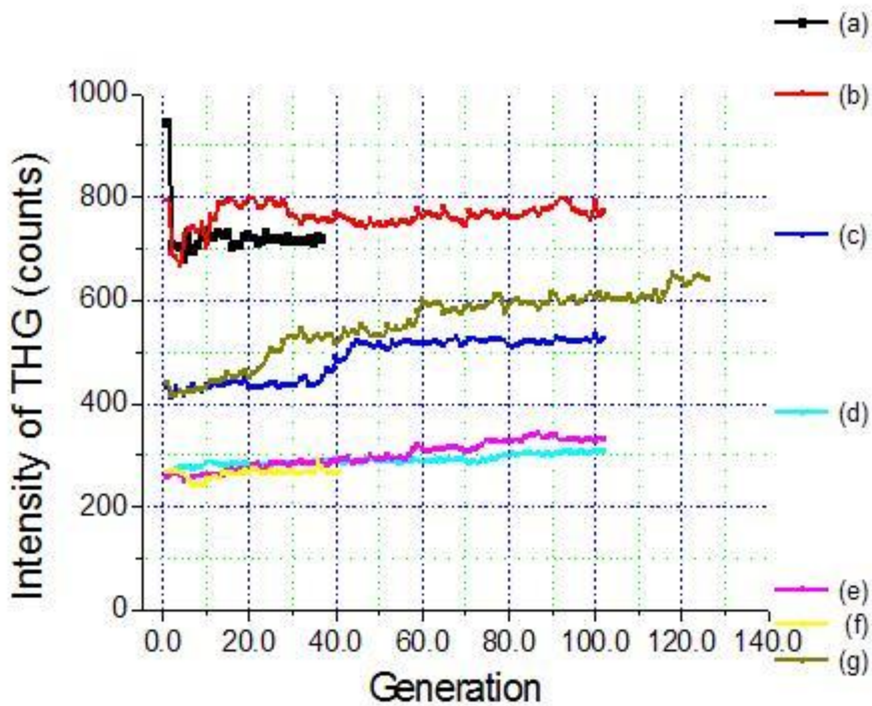

Fig. 5. The intensity of THG vs. generation # of the genetic algorithm in wavelength-control mode study

3.2.2.2 Test of the variation

We run genetic algorithm twice, with two different variations. One is using fixed variation 0.6. The other one, we just use 3% of the whole changeable range as the variation (Figs.6, 7).

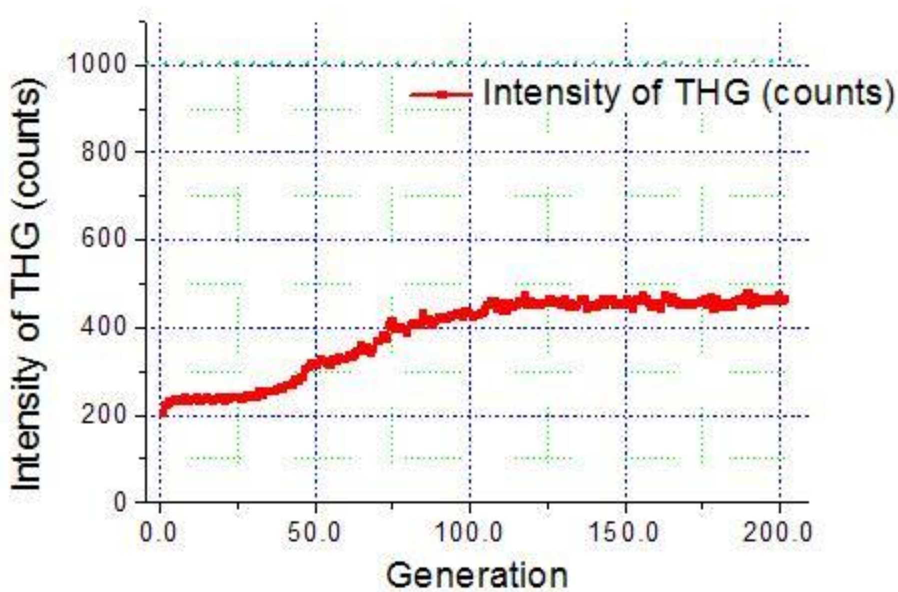

Fig. 6. The intensity of THG vs. generation # in wavelength-control mode with variation 0.6

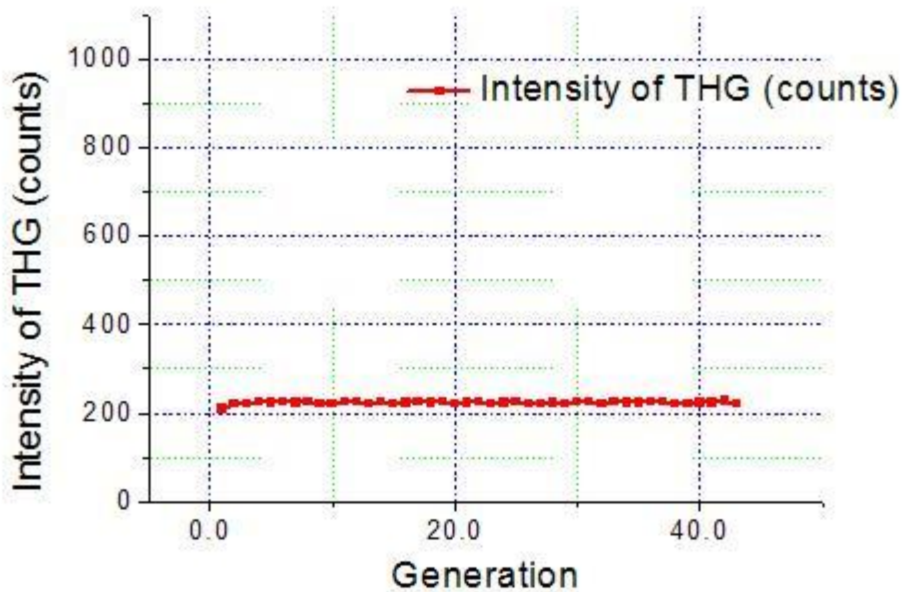

Fig. 7. The intensity of THG vs. generation # in wavelength-control mode with variation 3% of the whole changeable range

Both these two runs begin from an initial population consisting of 27 random individuals and 3 manually calculated individuals. These three manually added individuals have center wavelength 805nm, and delay 3500 fs. They are (1) with chirp $12000\ fs^2$ (2) with chirp $20000\ fs^2$ (3) with chirp $20000\ fs^2$ and quadratic chirp $-50000\ fs^3$

Both these two runs have the mutation rate of 0.2. In the first run, we set the variation equal to 0.6. We can see that the intensity of the THG increases as the generation number increases. Then we set the variation equal to 3% of the whole variable range. We can see that there is no improvement of the THG signal. This means that, when variation is 3% of the range, the genetic algorithm fails.

3.2.3 Genetic algorithm for the polynomial control

We let the genetic algorithm deal with the 4 parameters of the polynomial phase function, which are delay a_1, chirp a_2, quadratic chirp a_3, cubic chirp a_4. These four parameters then form the string structure, which is used as the individual of the genetic algorithm. The THG intensity is then used as the feedback for the genetic algorithm (Fig.8).

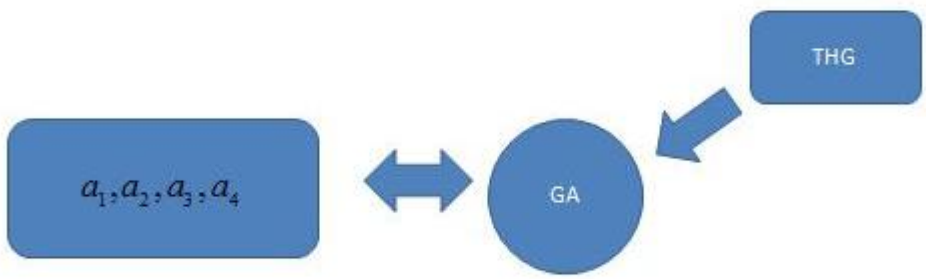

Fig. 8. A scheme for the genetic algorithm running in polynomial-control mode

3.2.3.1 Test of the random initial phase

We created the first population randomly and ran the genetic algorithm with the variation of 0.03 (Fig.9).

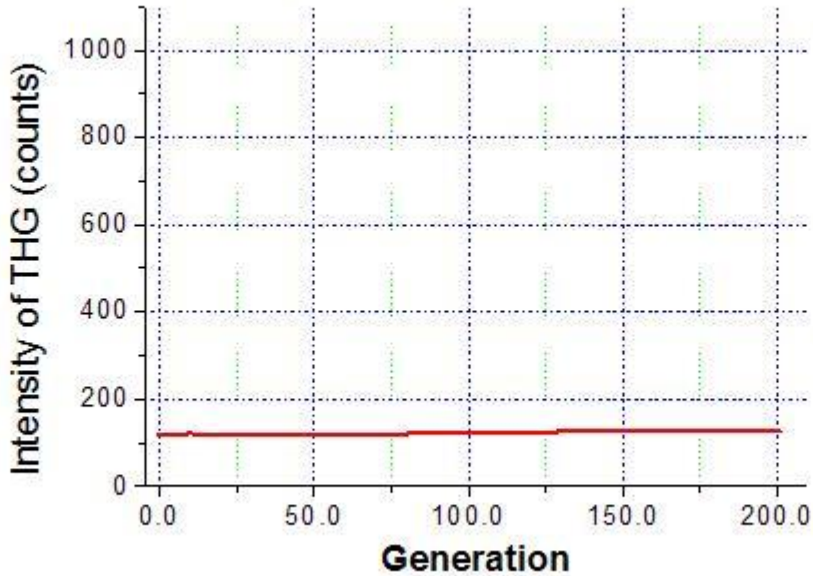

Fig. 9. The intensity of THG vs. generation # in polynomial-control mode without any initial seed individual

We can see that there is no improvement of the THG signal as the generation number increases. The genetic algorithm fails in this part of the test. We also test the behavior of the 4 parameters, and find that the changes of these 4 parameters are limited (Fig.10).

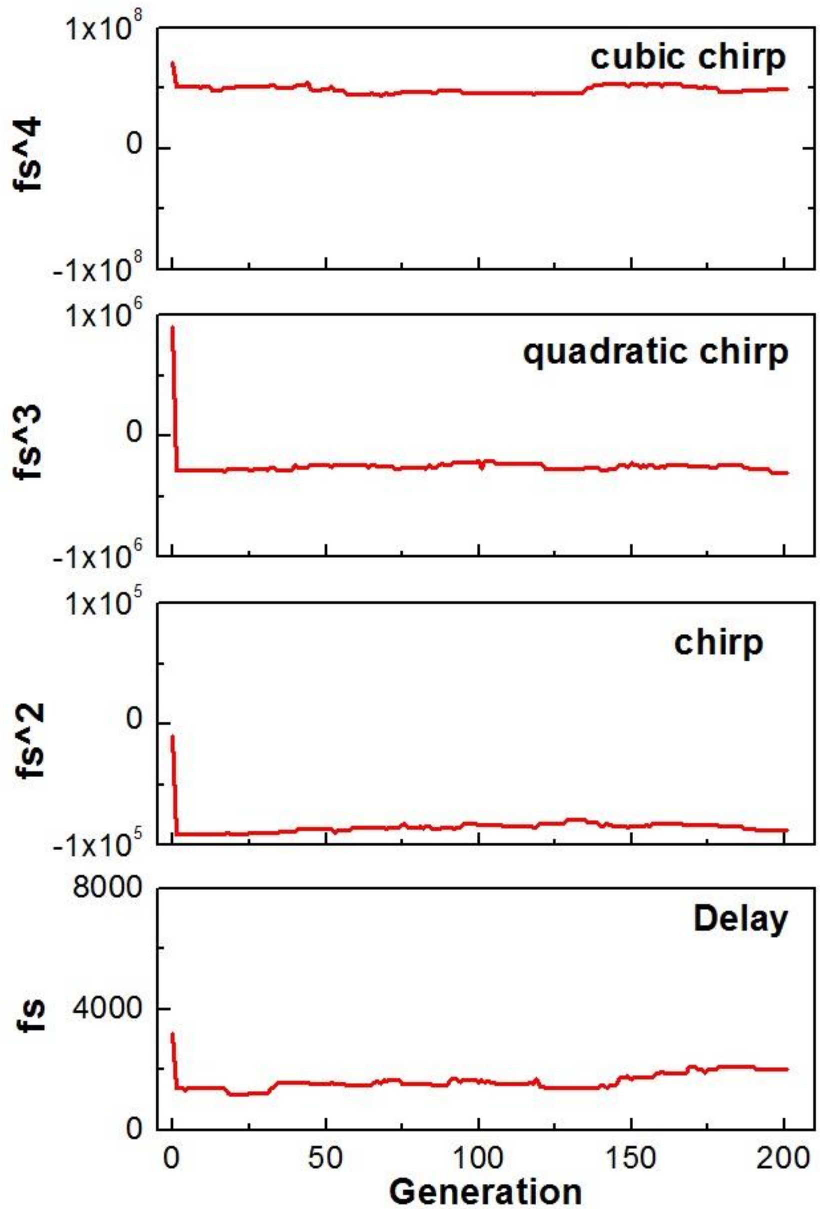

Fig. 10. The four polynomial parameters vs. generation # in polynomial-control mode without any initial seed individual

The failure of the genetic algorithm in this part again verifies that the genetic algorithm needs some signal to start with.

3.2.3.2 Test of the manually chosen phase as the initial phase

Then we add a manually optimized seed into the randomly created initial population. Let the genetic algorithm have some THG signal to start optimize.

We run this part of the experiment three times. The first time, we insert the manually created seed, with delay 3500fs and chirp 15000 fs^2 (Fig.11). The second time, the manually created seed has parameters: delay=3500fs and chirp=12000 fs^2 (Fig.12). The third time, the manually created seed has parameters: delay=3500fs and chirp= 28000 fs^2 (Fig.13). The genetic algorithm runs with the variation of 0.03.

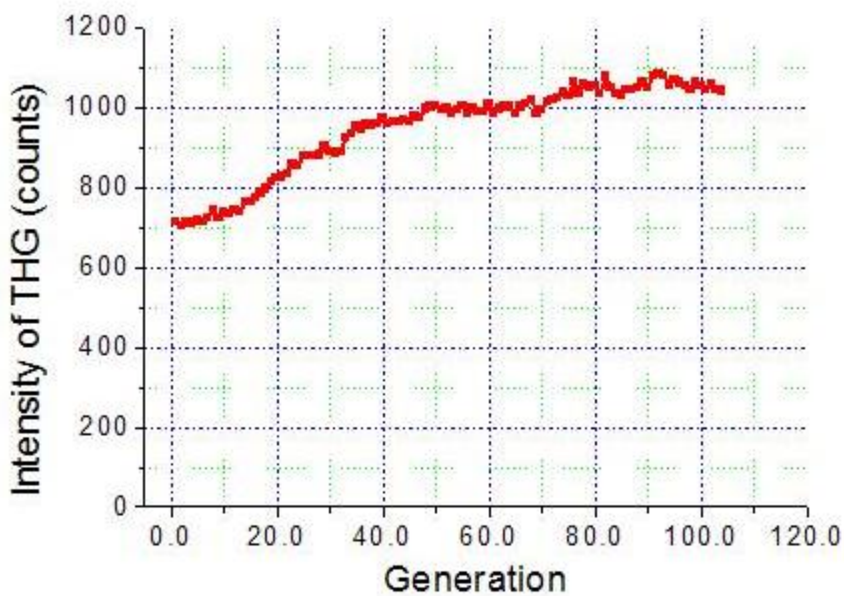

Fig. 11. The intensity of THG vs. generation # in polynomial-control mode with an initial seed individual with 15000 fs^2 chirp

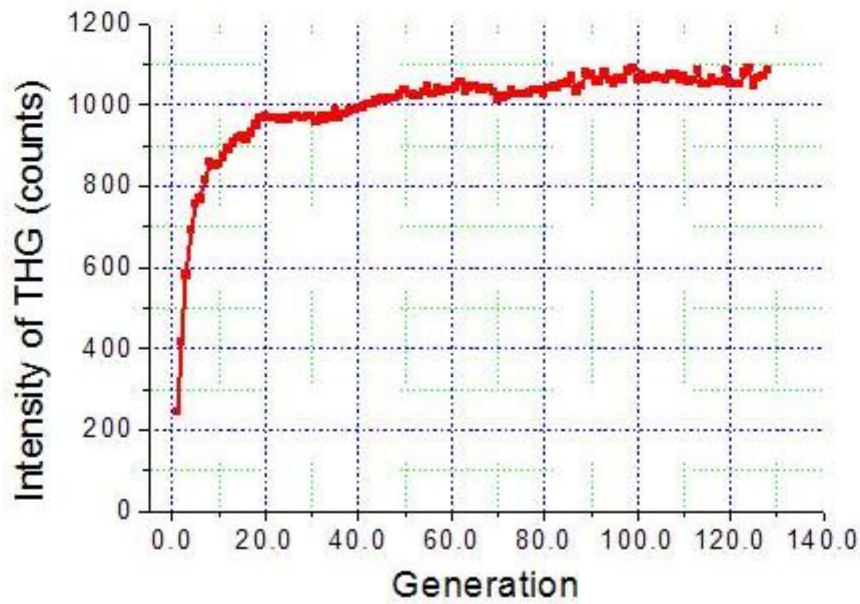

Fig. 12. The intensity of THG vs. generation # in polynomial-control mode with an initial seed individual with 12000 fs^2 chirp

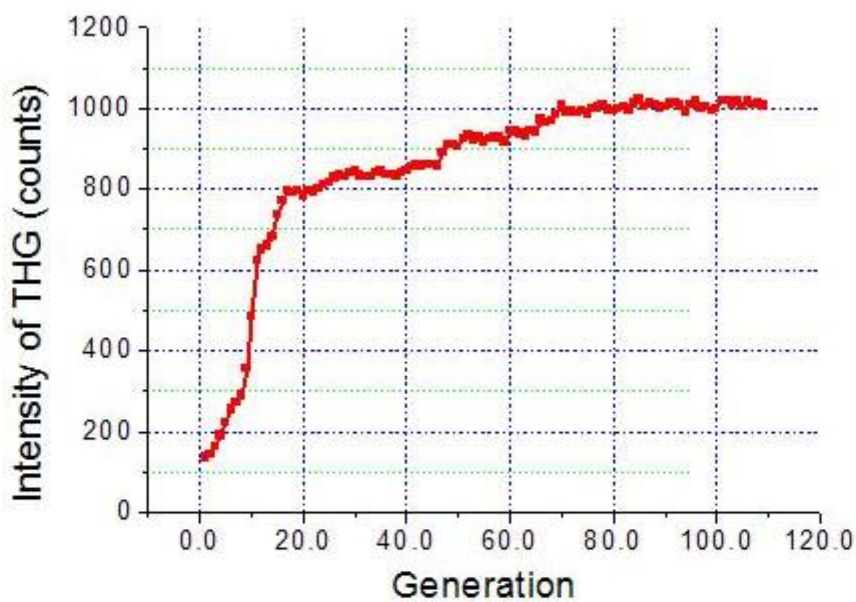

Fig. 13. The intensity of THG vs. generation # in polynomial-control mode with an initial seed individual with 28000 fs^2 chirp

The THG signal increased from 700 counts to 1030 counts for the first run, which is an improvement of 57%. The genetic algorithm converged after about 40 generations. In the second run, the THG signal increased from 245 counts to 1070 counts, which is an improvement by 5 times. The genetic algorithm converged after about 20 generation. In the third run, the THG signal increased from 136 counts to 1000 counts, which is an improvement by 8 times. The genetic algorithm converged after about 20 generations. All these three runs give us significant improvements of the THG signals, which is the result we expected.

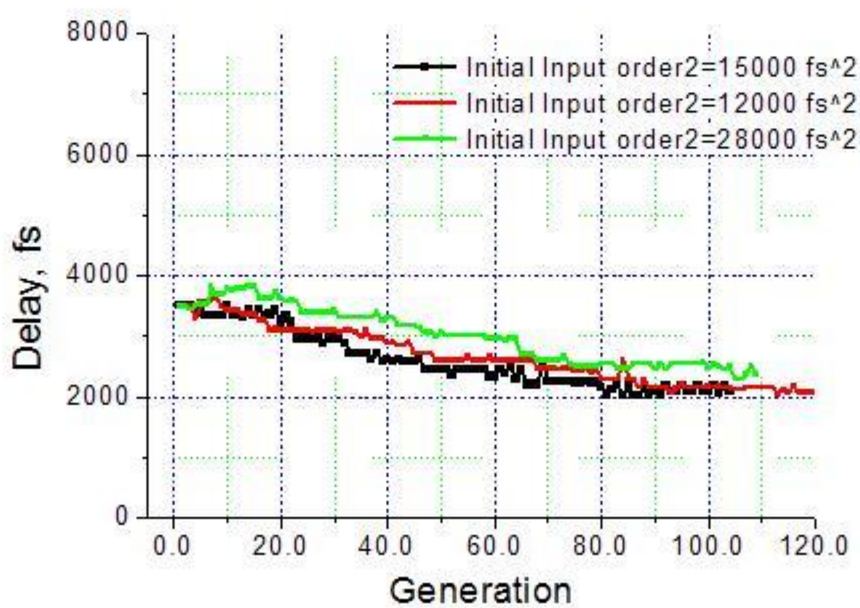

Fig. 14. The delay vs. generation # in polynomial-control mode with different initial input chirps

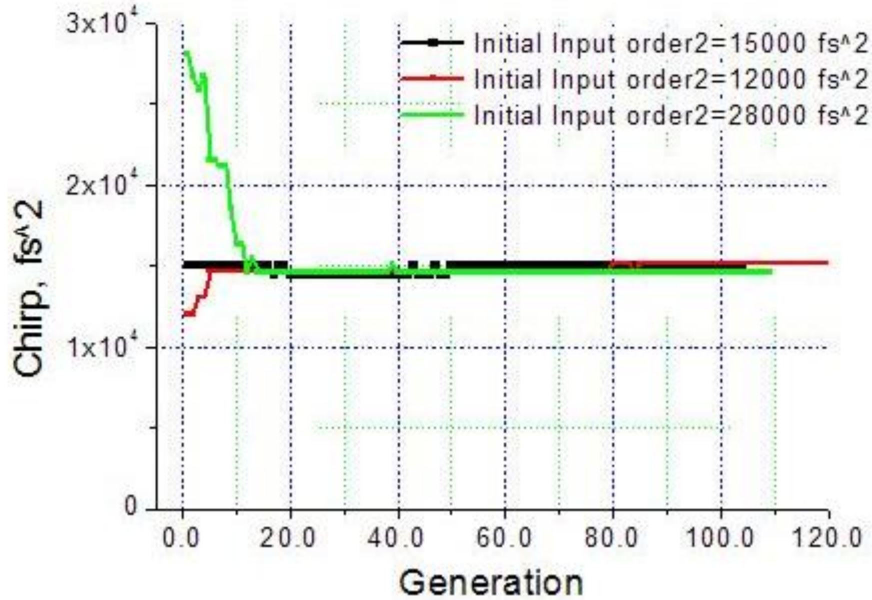

Fig. 15. The chirp vs. generation # in polynomial-control mode with different initial input chirps

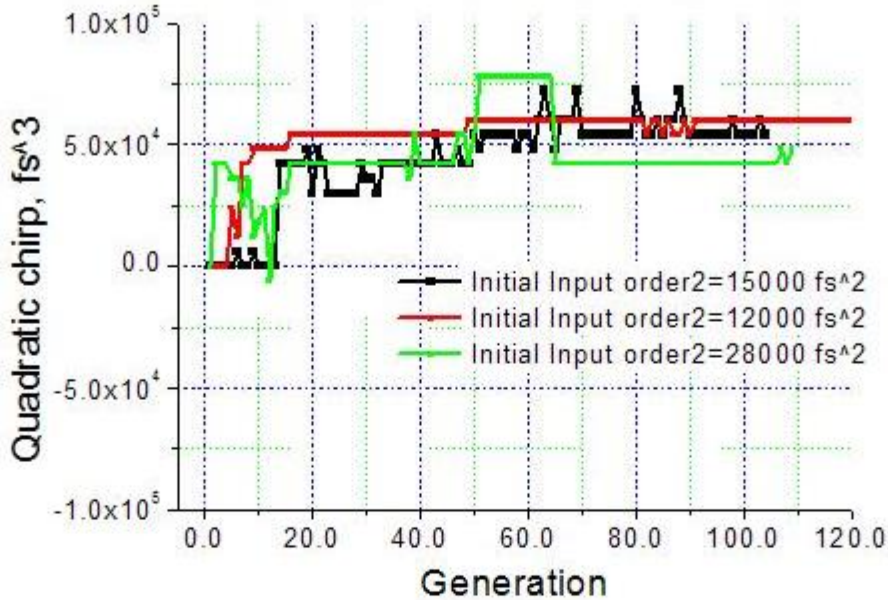

Fig. 16. The quadratic chirp vs. generation # in polynomial-control mode with different initial input chirps

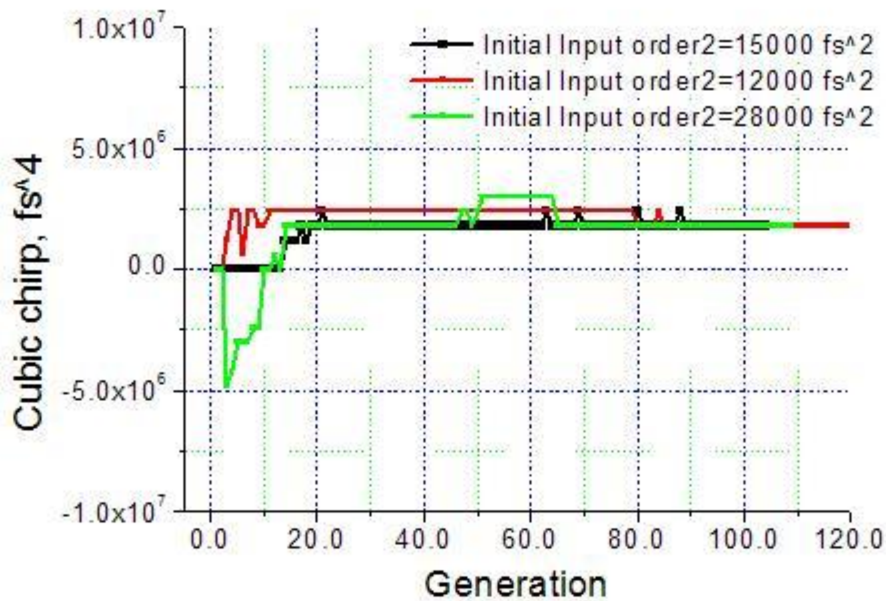

Fig. 17. The cubic chirp vs. generation # in polynomial-control mode with different initial input chirps

Then we check the behavior of the four parameters: delay, chirp, quadratic chirp and cubic chirp (Figs.14, 15, 16, 17). These four parameters converge as the generation number increases.

3.2.3.3 Test of the variation

We use a seed individual with delay=3500 fs and chirp=12000 fs^2. We choose different variations in this part of the experiment. In the first run, we choose a variation of 0.01. In the second run, we choose variation=0.03. In the third run, we choose variation=0.1. The result shows that the genetic algorithm running with variation=0.03

has the fastest convergence speed. Both the smaller variation 0.01 and the larger variation 0.1 have slower convergence speeds (Fig.18).

This result validates the discussion in section 2.4. When a genetic algorithm searches near the optimized point, a larger variation would cause the genetic algorithm searching far away from the optimized point. If the genetic algorithm searches not near enough to the optimized point, a small variation would have a slower convergence speed than the proper one.

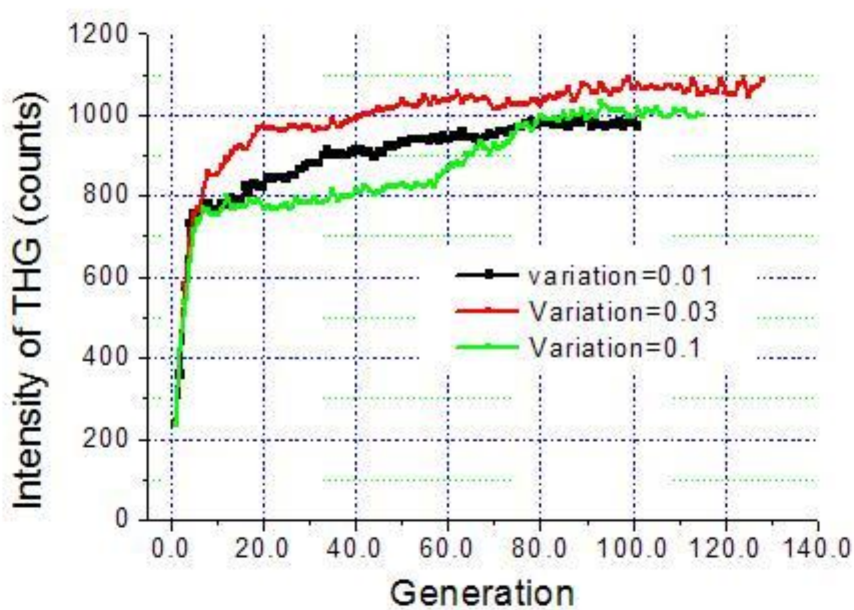

Fig. 18. The intensity of THG vs. generation # in polynomial-control mode with different variations

Then we check the behavior of the 4 parameters again (Figs.19, 20, 21, 22). We can see clearly that the 4 parameters in the genetic algorithm with variation 0.03 change steadily with a fast convergence speed. The 4 parameters in the genetic algorithm with variation=0.01 change steadily with a slower convergence speed. The 4 parameters in the genetic algorithm with variation=0.1 jump as the generation number increases.

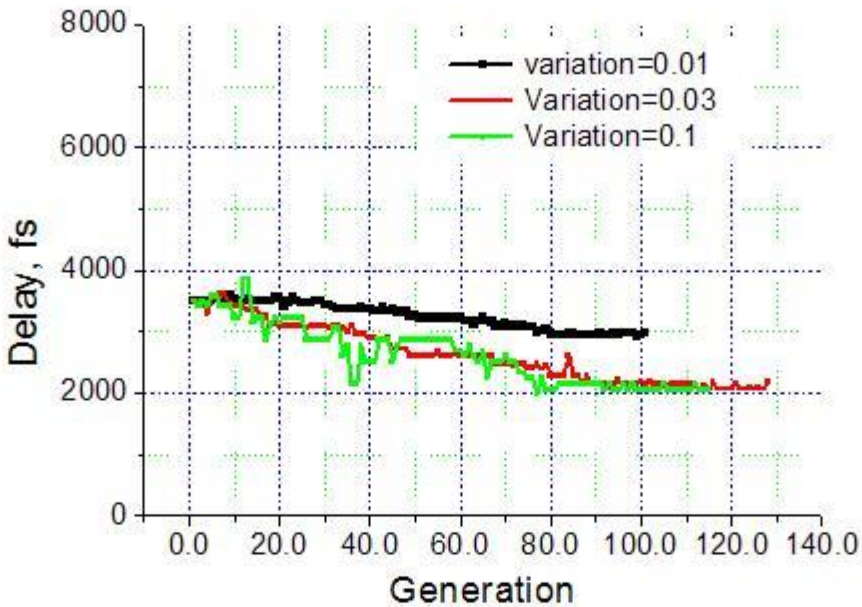

Fig. 19. The delay vs. generation # in polynomial-control mode with different variations

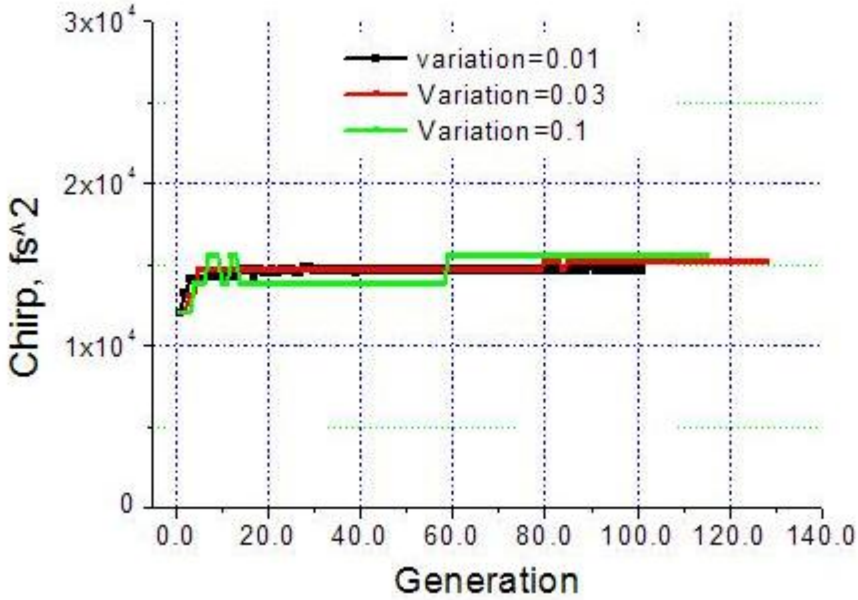

Fig. 20. The chirp vs. generation # in polynomial-control mode with different variations

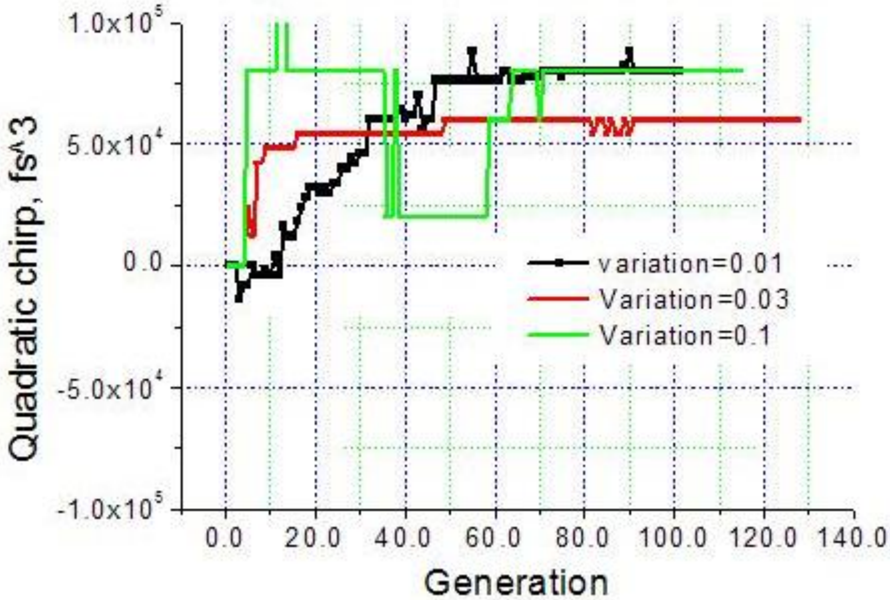

Fig. 21. The quadratic chirp vs. generation # in polynomial-control mode with different variations

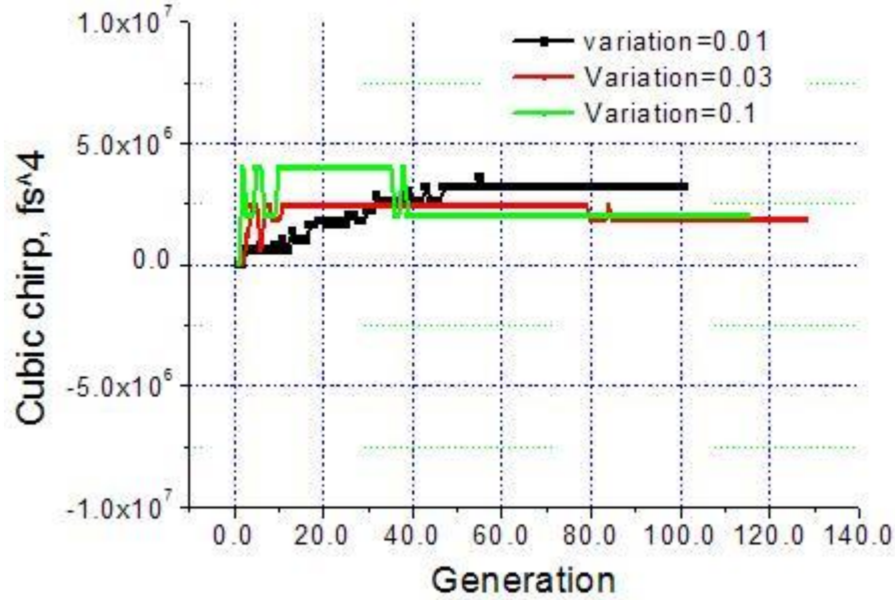

Fig. 22. The cubic chirp vs. generation # in polynomial-control mode with different variations

4. RESULTS AND SUMMARY

The genetic algorithm gives us expected results in the polynomial-control part of the experiment. The THG signal can increase by as much as a factor of 8, if the genetic algorithm starts from a lower THG signal.

The genetic algorithm fails in the wavelength-control mode. There are two possible reasons. One is that the noise is too large. The noise of the measurement of the THG signal is about 10% in the wavelength-control mode, so the genetic algorithm cannot compare which individual is good in one generation. Reducing the noise of the measurement is one possible way to let the genetic algorithm in wavelength-control mode work. The other possible reason that the genetic algorithm fails is that the THG signal is sensitive to continuous phase function. The phase function we give in wavelength-control mode is a descrete phase function. In order to solve this problem, we can use interpolation to smooth the descrete phase function, make it continuous.

Another result we get from the above experiment is that the variation should neither be too large nor too small. A large variation could cause the individual changes far away from the optimized point. A small variation could slow the convergence speed. We need to choose an appropriate variation for the THG signal.

REFERENCES

1. D. Zeidler, S. Frey, K.-L. Kompa, and M. Motzkus, Phys. Rev. A **64**, 023420 (2001).
2. Y. Huang and A. Dogariu, Opt. Express **14**, 10089-10094 (2006).
3. N. Dudovich, D. Oron, and Y. Silberberg, Nature **418**, 512-514 (2002).
4. D. Oron, N. Dudovich, D. Yelin, and Y. Silberberg, Phys. Rev. A **65**, 043408 (2002).
5. R. Morita, M. Yamashita, A. Suguro, and H. Shigekawa, Opt. Commun. **197**, 73-81 (2001).
6. J.P. Ogilvie, D. Debarre, X. Solinas, J.L. Martin, E. Beaurepaire, and M. Joffre, Opt. Express **14**, 759-766 (2006).
7. S. Zhang, L. Zhang, X. Zhang, L. Ding, G. Chen, Z. Sun, Z. Wang, Chemical Physics Letters **433** 416-421 (2007).
8. X. Li, H. Zhang, X. Zhang, S. Zhang, G. Chen, Z. Wang, Z. Sun, Chin. Phys. Lett. **25** 528-531 (2008)
9. A.M. Weiner, Rev. Sci. Instrum. **71**, 1929 (2000).
10. C. W. Hillegas, J. X. Tull, D. Goswami, D. Strickland, and W. S. Warren, Opt. Lett. **19**, 737-739 (1994).
11. A. M. Weiner, D. E. Leaird, J. S. Patel, and J. R. Wullert, Opt. Lett. **15**, 326-328 (1990).
12. M. M. Wefers and K. A. Nelson, Opt. Lett. **18**, 2032-2034 (1993).
13. M. M. Wefers and K. A. Nelson, Opt. Lett. **20**, 1047-1049 (1995).

14. A. M. Weiner, D. E. Leaird, J. S. Patel, and J. R. Wullert II, IEEE J. of Quantum Electronics, **28**, 908-920 (1992).

15. M. M. Wefers and K. A. Nelson, IEEE J. of Quantum Electronics, **32**, 161-172, (1996).

16. D. E. Goldberg, *Genetic algorithms in search, optimization, and machine learning* (Addison-Wesley Pub. Co., Reading, MA, 1989).

17. P. Tournois, Optics Communications, **140**, 245-249, (1997).

VITA

Name:	Hua, Xia
Address:	Physics Department, c/o Dr. Alexei Sokolov, Texas A&M University College Station, TX 77843-4242
Email Address:	huaxia@neo.tamu.edu

Education:	B.S., Physics, Wuhan University, 2007 B.Eng., Computer Science & Technology, Huazhong University of Science & Technology, 2007